LA-UR-15-26344
Approved for public release; distribution is unlimited.

Title: A Four Channel Beam Current Monitor Data Acquisition System Using Embedded Processors

Author(s): Wheat, Robert Mitchell Jr.
Dalmas, Dale A
Dale, Gregory E

Intended for: Report

Issued: 2015-08-11

A Four Channel Beam Current Monitor Data Acquisition System Using Embedded Processors

Robert M. Wheat, Dale A. Dalmas, and Gregory E. Dale

Introduction

Data acquisition from multiple beam current monitors is required for electron accelerator production of Mo-99. A two channel system capable of recording data from two beam current monitors has been developed, is currently in use, and is discussed below. The development of a cost-effective method of extending this system to more than two channels and integrating of these measurements into an accelerator control system is the main focus of this report. Data from these current monitors is digitized, processed, and stored by a digital data acquisition system. Limitations and drawbacks with the currently deployed digital data acquisition system have been identified as have been potential solutions, or at least improvements, to these problems. This report will discuss and document the efforts we've made in improving the flexibility and lowering the cost of the data acquisition system while maintaining the minimum requirements.

Figure 1- The currently deployed Beam Current Monitor acquisition system.

The Currently Deployed Acquisition System

The currently deployed digital data acquisition system for the beam current monitors is a PXI Chassis System from National Instruments with a digitizing card and a CPU module installed. The Chassis is PXI-1042 with a PXI-8186 CPU Controller and a PXI-5122 Oscilloscope plugin performing the actual data conversion and digitization. Software has been developed with National Instruments LabView to control the hardware, acquire and process the data, and save data to disk at predetermined intervals. The National Instruments PXI crate

solution provides 2 channels from the single PXI-5122 plug-in, sampling at a rate of up to 100 MS/s. A picture of the current PXI crate system is shown in Figure 1.

Typical accelerator applications run beam pulses at up to 100 Hz, and perhaps faster in the future. The currently deployed PXI Chassis system is not fast enough to acquire all pulses, and certainly if additional channels of acquisition were added this would still be the case. Currently there is an external frequency counter providing the “instantaneous” repetition rate of the system to be recorded at the same time and interval as raw data such that beam history can be recreated without having to save huge amounts of data. This same procedure of saving data at preset intervals can and should be applied to the consideration of any other acquisition system.

The PXI crate allows for expansion in the form of adding new modules or plug-ins, at least until the chassis is full. The currently deployed chassis will allow 6 more single wide modules or plug-ins to be added to the system. The controller plug-in is an embedded type computer, using Intel processors typically, and running Windows and Labview. The embedded computer also has a laptop style hard drive for data persistence, but data must be moved after an irradiation run since the computer is part of the acquisition system. National Instruments also manufactures the 5122 card in a PCI style, which can plug into a desktop type computer; however the card is very large and will not plug into just any desktop computer. Figure 2 shows a picture of the PCI version of the 5122 Oscilloscope card.

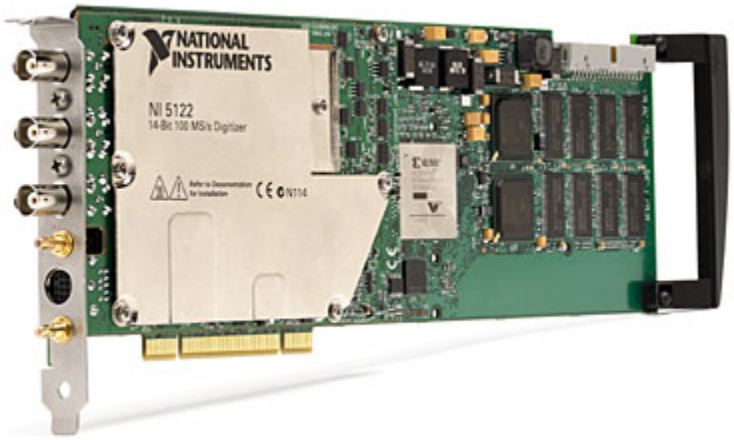

Figure 2 - A picture of the PCI version of the National Instruments 5122 Oscilloscope card.

Some of the inconveniences of a system as has been deployed here include the fact that the computer, including the control software and data disk drive, is a part of the acquisition system, and to control the system remotely one must use something like Remote Desktop or VNC. In this case it’s “all or nothing”. Observing the manner in which modern day accelerator systems are interfaced and controlled it becomes obvious that this paradigm can be improved. Specifically, using EPICS and the IOC (Input Output Controller) type configuration the Beam Current Monitor acquisition system can easily be turned into an IOC, publishing its data on the internal controls network, and looking for controlling messages on the same network.

For many reasons including cost, convenience, ease of use, maintainability, etc. it was decided to review all options before simply proceeding with a software rewrite to adopt the new IOC type paradigm. For example, the National Instruments hardware is fairly expensive and there exists the possibility of needing to replicate many channels of acquisition in a proposed Moly99 irradiation facility, therefore, cost per channel is definitely an issue. Also a consideration is the potential lifetime of the hardware chosen such that getting stuck with legacy hardware is prevented or at least minimized.

Directions and Considerations

There are many available commercial digital data digitization and acquisition systems available for costs ranging from a few hundred dollars per channel to several thousand dollars per channel. Obviously, cost is one of the considerations. Some of the requirements for any channel of acquisition include a sample rate of at least 50 MS/s but 100 MS/s is desirable. There is a requirement for at least 512 points in a dataset, but 1024 is desirable. Finally, a resolution of at least 8 bits is required but more is better.

A fairly exhaustive search for hardware options was performed keeping the following details in mind, but allowing for slight deviations for completeness:

- Sample rate to be at least 50 MS/s
- Bits of resolution at least 8
- Cost to be minimized
- One acquired data set to be about 10 microseconds in length
- Maintainability
- Programmability

Table 1 provides the results of the hardware search performed. The National Instruments hardware was included since it has been the baseline since the beginning of the Beam Current Monitor. The Measurement Computing hardware was included in this table and report even though the minimum requirements were not met since Measurement Computing is considered one of National Instruments competitors, and it seemed only fair to include that here.

Table 1- Comparison of some of the acquisition system replacement options.

Name	Model Numbers	Description	Important Specs	Cost	Cost/ Channel
National Instruments	PXIe-1078 (1 ea)	PXI Crate or Mainframe	9 Slot 3U PXI Express	$2,408.00	
	PXIe-8840 (1 ea)	Embedded Controller	Core i5-4400E	$4,660.00	
	PXI-5122 (2 ea)	O-Scope Module	256 MB/ch, 14-bit, 100 MS/s	$19,196.00	
				$26,264.00	**$6,566.00**
National Instruments	PCI-5122 (2 ea)	O-Scope Module	32 MB, 14-bit, 100 MS/s	$16,498.00	
Dell	Precision 5810	Control Computer	16 GB, Xeon 6 Core	$2,599.00	

				$19,097.00	**$4,774.25**
Measurement Computing	USB-2020 (2 ea)	12-Bit, 20 MS/s, USB Board	20 MS/s is actually unacceptably low.	$1,998.00	
Dell	Precision 5810	Control Computer	16 GB, Xeon 6 Core	$2,599.00	
				$4,597.00	**$1,149.25**
Signatec	pX14400D (2 ea)	PCIe Bus, DC-Coupled	400 MS/s, 14 bit	$14,300.00	
Dell	Precision 5810	Control Computer	16 GB, Xeon 6 Core	$2599.00	
				$16,899.00	**$4,224.75**
Linear Tech Demo Hardware	DC1732B-AB	LTM9012 Quad uModule ADC + Driver	14 Bit, 125 MS/s	$300.00	
	DC1075A	Clock Divider Board		$50.00	
	DC1371A	USB Acquisition Controller		$1,950.00	
Dell	Precision 5810	Control Computer	16 GB, Xeon 6 Core	$2,599.00	
				$4,899.00	**$1,224.75**
AlazarTech	ATS9440-002	PCIe Digitizer	14 bit, 125 MS/s, 4 Ch, 65 MHz BW	$5,495.00	
Dell	Precision 5810	Control Computer	16 GB, Xeon 6 Core	$2,599.00	
				$8,094.00	**$2,023.50**
OpenADC/Digilent	OPENADC-02 (4 ea)	OpenADC Board	10 bit, 105 MS/s	$548.00	
	Nexys3 (4 ea)	FPGA Development Board	Spartan-6	$1,236.04	
Raspberry Pi	Raspberry Pi 2 Model B	Control Computer, IOC	900 MHZ Quad-core Arm Cortex-A7	$40.00	
				$1,824.04	**$456.01**

Again, the Measurement Computing device does not meet the minimum requirements because of sample rate. Therefore, it was not seriously considered as a potential solution. The National Instruments option, while providing the current "baseline", has a few limitations, besides the cost, that have been mentioned already. Some of these limitations, in particular the use of a "regular" computer as the IOC, are also shared with the Signatec and the Alazar Tech options as well, since a PCIe expansion slot is a requirement. The Dell computer chosen in Table 1, for several of the options requiring a computer, was the mid-range version workstation from Dell. An actual application would define a more or less capable computer. In addition to the added cost, a "regular" computer, although it can be set up and operated using the IOC philosophy described below, requires much more space and additional administration time and effort when compared to an embedded solution. The two best choices from Table 1 for potential BCM acquisition system hardware are the Linear Tech ADC hardware and the OpenADC interfaced to the Digilent FPGA combination.

The Current Choice of Hardware

Currently we have chosen to develop a system using the OpenADC board interfaced to an FPGA development board. Although cost is slightly better for the OpenADC option as compared to the Linear Tech option, this was not the only reason the OpenADC option was chosen. The fact that some software as well as some HDL was provided with the OpenADC option, as well as the fact that all of the board design information for the OpenADC unit itself was publically available, with no proprietary licensing issues, were the deciding factors in favor of the OpenADC option. Board design availability was important for two reasons, at least; the first being that when changes to the circuit are made, board redesigns will be easy and straight forward, and the second being that when it comes time to replicate the acquisition systems many times over, this too will be simple, straight forward, and inexpensive. The OpenADC boards with the LANL modifications are excellent choices for data acquisition in a high volume, rapid throughput production system for several reasons. The boards are very robust, insensitive to environmental thermal fluxes, static discharges, etc. In addition the ADC delivers the data over a parallel bus maximizing the throughput by providing all the bits of sampled data in a single ADC read operation. Finally, controlling communications with an FPGA provides for maximizing communications speeds as well as allowing growth potential, system improvements, and additional channels.

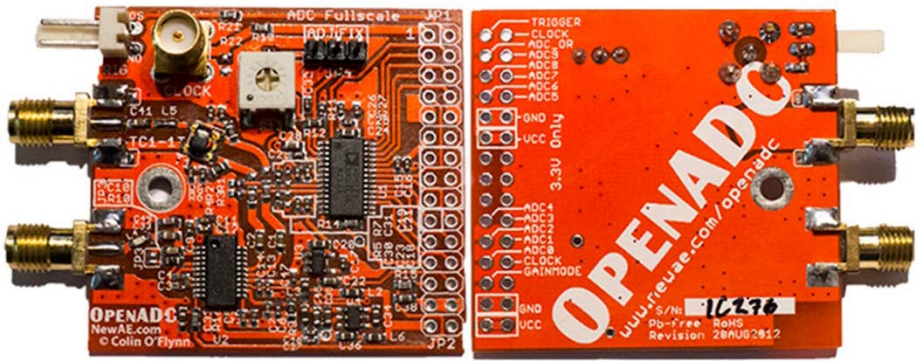

Figure 3- Front and back of the OpenADC board.

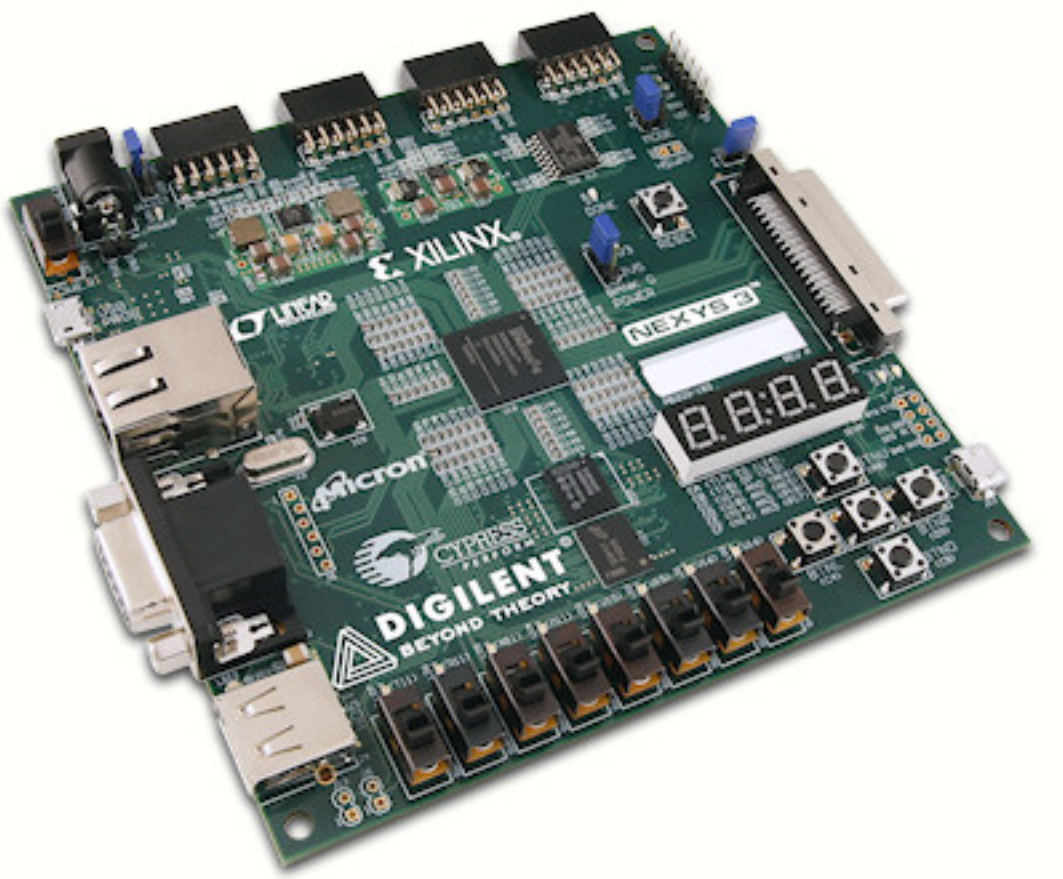

Figure 4- The Digilent Nexys 3 FPGA development board.

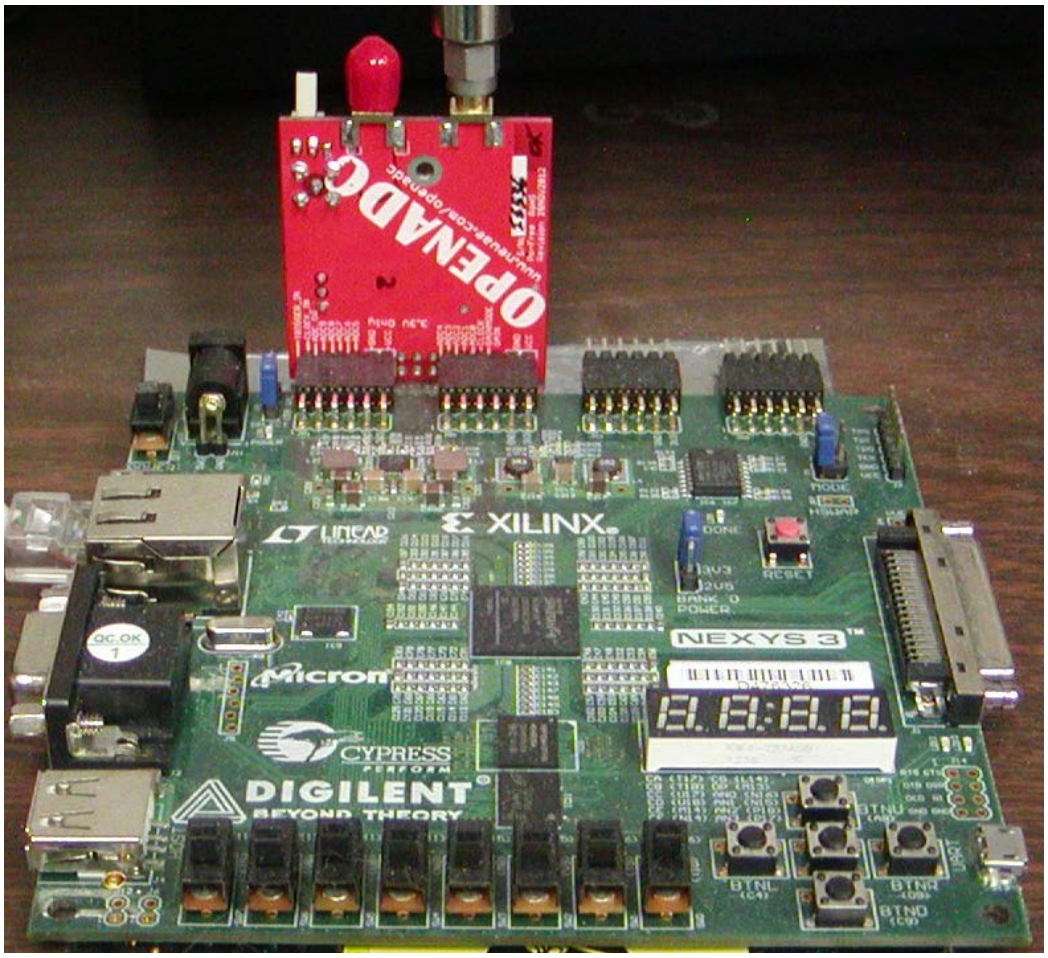

Figure 5- The OpenADC board interfaced to the Digilent board through two of the Digilent board's peripheral module connectors.

Configuration

The hardware is currently configured with one channel of ADC driven or controlled by one FPGA development board. Figures 3, 4, & 5 show the OpenADC board, the Digilent Nexys 3 development board, and the two boards interfaced together, respectively.

Although the current configuration interfaces a single ADC board to a single FPGA board, each FPGA board could easily control 5 ADC boards, and more if a custom FPGA board were designed. For ease of debug and diagnostics, however, this early stage design remains simple using the one-to-one ratio of ADC to FPGA.

The IOC Controller

An IOC (Input Output Controller) as defined for this acquisition system comprises all of the hardware necessary to interface to and control the appropriate sensors as well as acquire data from these sensors. In this case, the Beam Current Monitor IOC records data from the Current Transformers and posts the data acquired to a location accessible over a network and in a form usable and meaningful to the user while at the same time monitoring for commanding signals to change configuration or behavior. Stated another way, an IOC is the following:

- Hardware to interface to and control sensors
- Data acquisition system
- Data processor
- Network client (not server)

To combine multiple copies of the FPGA/ADC combinations which in turn are each connected to a sensor, some sort of computer controller is used to orchestrate the entire assembly. The computer controller chosen to make the entire assembly behave as an integrated IOC is the Raspberry Pi 2. The Raspberry Pi 2 runs a version of Linux, Raspbian based on Debian, and has standard interface ports such as Ethernet and USB to allow for connections to other hardware and to a network. In addition, the entire operating system with the control software and the EPICS software as well all fits on an SD Card which can easily be replicated based on a "central" standard repository and distributed to as many IOC's as is required. With written instructions as well as the occasional use of the large user community, implementation, operation, and support of both the operating system and the EPICS software will be easily performed by any mid-level IT person.

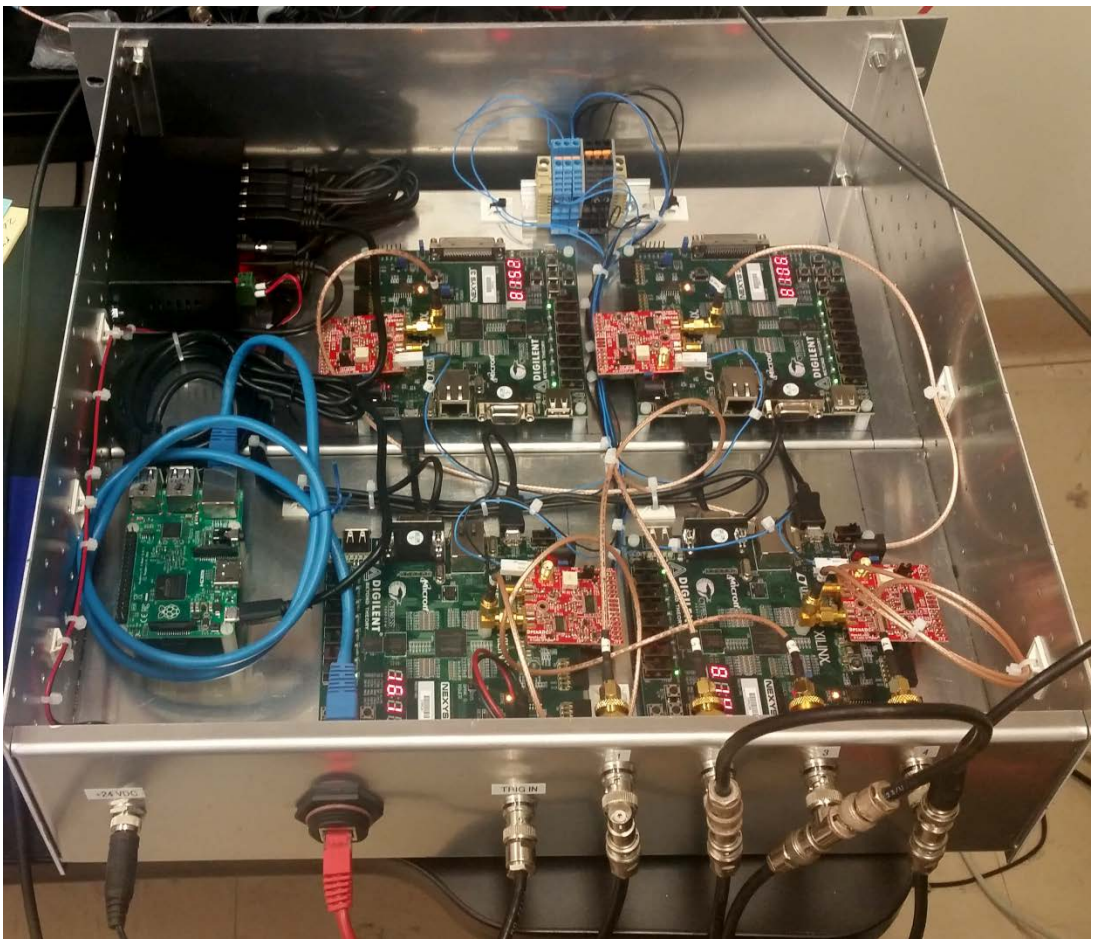

Figure 6- The four channel acquisition system or IOC.

The "Hardware to interface to and control sensors" in this case is 4 Digilent Nexys 3 development boards, each with a single OpenADC board plugged into two of the PMod

(Peripheral Module) connectors, and one Raspberry Pi embedded controller. The assembly of these components is shown in the Figure 6 picture, and a block diagram of the assembly in Figure 7. The Raspberry Pi runs software developed for this purpose which controls each of the FPGA boards and their associated ADC boards, acquiring data when ready and "posting" this data to the EPICS server database (more about the EPICS server coming up next). In this case, communicating with the FPGA, the Raspberry Pi performs the "Data acquisition" functionality. Once the data has been acquired by the Raspberry Pi any amount of "Data processing" can occur prior to the data being "posted" to the EPICS server. For now, the specifics of the data processing performed by the Raspberry Pi have not been fully defined, and the definition of the data processing tasks turns out to have deep philosophical connotations. Finally, the Raspberry Pi takes care of "posting" the data to the EPICS server, as a "Network Client", while constantly monitoring the server for commands to change settings such as Hi/Lo gain settings, variable gain voltage settings, and pre-trigger time allowance, also as a "Network Client".

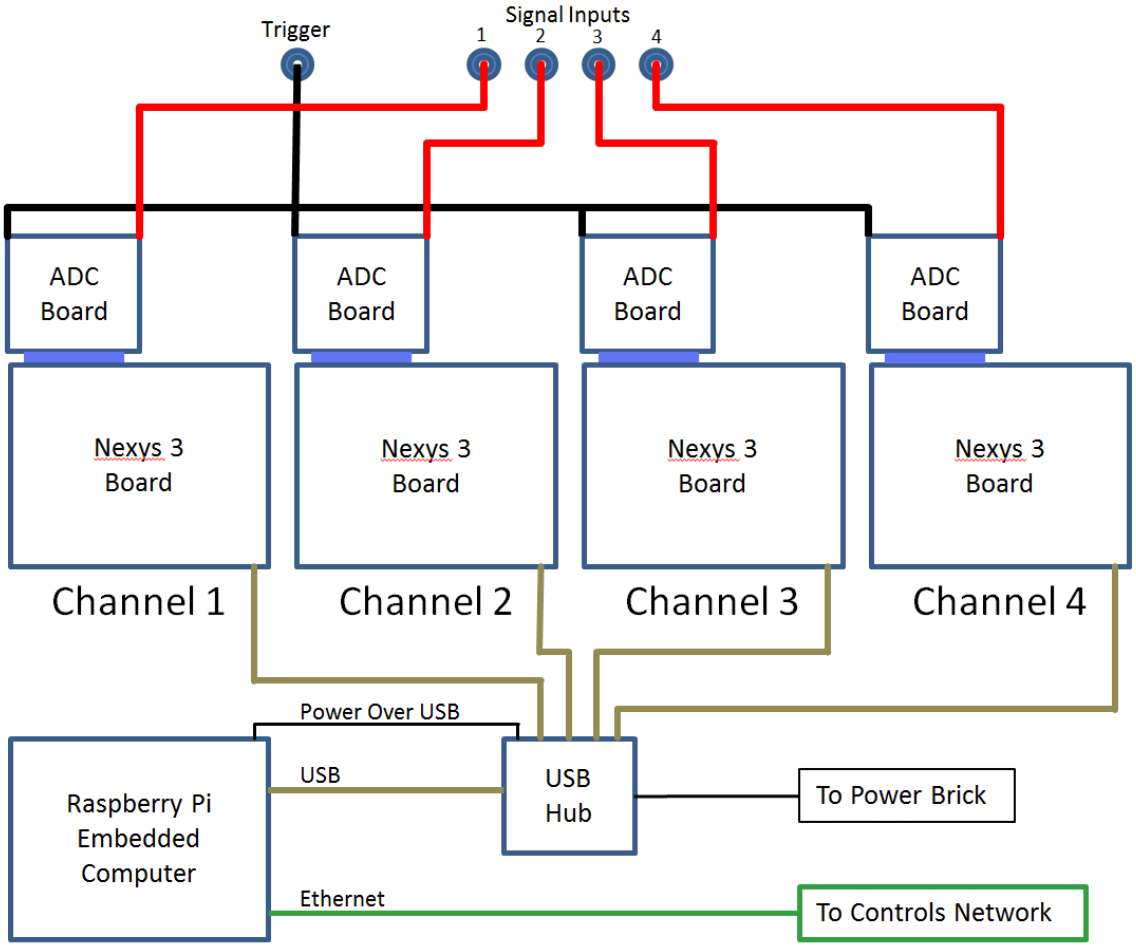

Figure 7 - A block diagram of the four channel acquisition system or IOC.

The part of the control system that allows the "connection" of a user to the IOC is the EPICS part of the system. EPICS fills the network server role, accepting input from a user which is turned into commands to control the sensors or the behavior of the sensor acquisition system, to answer to user requests for data, and to store the data provided by the acquisition system until either replaced by new data or forwarded on to a requesting user.

The Software

There are three distinct pieces to the software written for the Beam Current Monitor data acquisition system. There is the FPGA hardware description software, written in VHDL. There is the driver software written in 'C' and running on the Raspberry Pi, driving the FPGA and the ADC, collecting the data, and "posting" the data to the EPICS server. Finally, there's the GUI software, written in LabView, running on most any Windows computer that is connected to the same controls network to which the Raspberry Pi is connected, providing all of the specific "Beam Current Monitor Data Acquisition System" specific functionality including displays of the data plots and graphs, controls for the user, buttons to turn on and off any data persistence, and obviously on/off buttons to start and stop the acquisition system processes. The complexity of the software should be fairly low such that persons familiar with the use of laboratory equipment such as oscilloscopes, signal generators and such should have no problems setting up and using the software. The EPICS software, the Linux OS software, and any other software that is used as provided publicly is not considered custom for this application and is not included in this list.

The FPGA hardware description software is simple. It first provides signals to components of the ADC circuit to keep the acquisition alive and configured correctly, such as the variable voltage for the variable gain amplifier and the clock signal for the ADC itself. For communications with a controller the FPGA software provides a register based communication system "connected" to the USB port or bus. When the controlling computer, the Raspberry Pi, needs to adjust the configuration of an aspect of the ADC, the information is placed in the appropriate Register. In a like manner, when data has been acquired and is ready for transfer, and the computer controller has requested the data, the FPGA software makes the data available on the USB bus for the requesting controller.

The driver software running on the Raspberry Pi 2 embedded computer controller is the software that really turns the BCM Acquisition system chassis into an IOC, as explained earlier. The software, written in 'C', makes use of libraries from Digilent for High Speed USB communication with the Nexys 3 FPGA development board, libraries from the EPICS community to provide the capability of communication with the EPICS server, as well as normal 'C' system libraries.

Also running on the Raspberry Pi 2 is the EPICS server software providing the commanding to the driver software and accepting the data from the driver software. The EPICS server software "understands" the process variables involved with both the control aspects as well as the data acquisition aspects of the system to which it is connected, and monitors proper ports on the network for either requests for values of any of these PV's or for values for these PV's to post as current.

The third piece of software to make a system such as the BCM Data Acquisition system usable by people is the Graphical User Interface (GUI). Since the Raspberry Pi is running the EPICS server software and is monitoring the network, the GUI does not need to be running on the Raspberry Pi, but only needs to be running on a networked computer that has access to the same network on which the Raspberry Pi is residing. The GUI software should provide all of the controls that any user might need as well as data displays in the form of numeric displays, plot, graphs, dials, or whatever is appropriate.

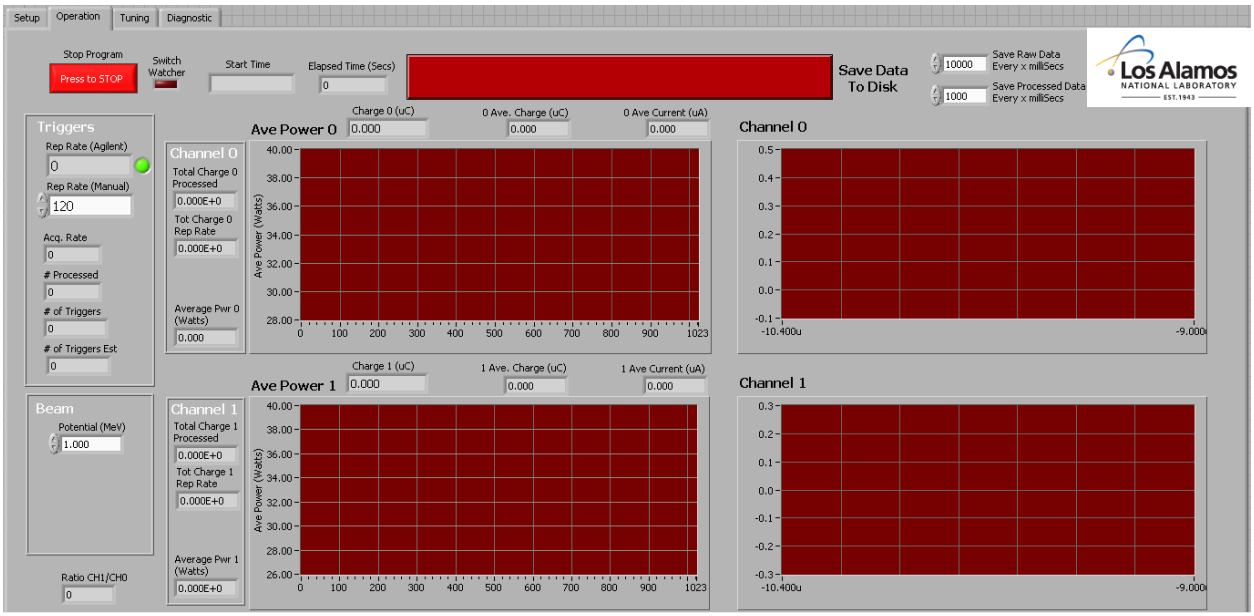

Figure 8 - A snapshot of a current version of the BCM GUI, for monitoring and controlling only two channels.

Figure 8 is a screen snapshot of a current version of the BCM GUI, showing the operations tab of the LabView Vi or Virtual Instrument (GUI). Included here are some controls as well as some data displays. The amount of data processing done by the GUI software is really defined by the developer prior to the writing of the software, just as it is with the "driver" software. Some of these decisions might affect the overall speed of the system, or the reliability of the system, or some other important aspect of overall system performance, so care and consideration should be involved when determining the "separation of duties" between the driver software and the GUI.

Modifications to the ADC Board

Some of the modifications made to the OpenADC board, as delivered, are shown in Figure 14 (note that the schematic in Figure 14 separates the AD8331 device into the Low Noise Amplifier (LNA) and the Variable Gain Amplifier (VGA) even though they come in the same 20-pin package) and include:

- Connectors for the FPGA interface
- Input cutoff frequency
- Cutoff frequency between VGA and ADC
- Input Impedance

For the connectors allowing the OpenADC board to plug into the PMod connectors of the Digilent Nexys 3 development board, right angle female header connectors were installed. These allowed, with the addition of male to male pin header adapters, the OpenADC board to be plugged into the Digilent boards in such a way as to sit parallel to the Digilent board. The OpenADC placement can be seen in Figure 6.

Since the OpenADC boards are specifically designed for very high frequency signals, the low frequency cutoff is set, from the factory, to such a high value that the signals expected from a Beam Current Monitor would show significant droop. From the factory this low frequency cutoff is modeled to be just under 32.0 kHz. A plot of the frequency response of the input portion of the OpenADC circuit as delivered from the factory is shown in Figure 9 while a schematic of the modeled circuit is shown in Figure 10.

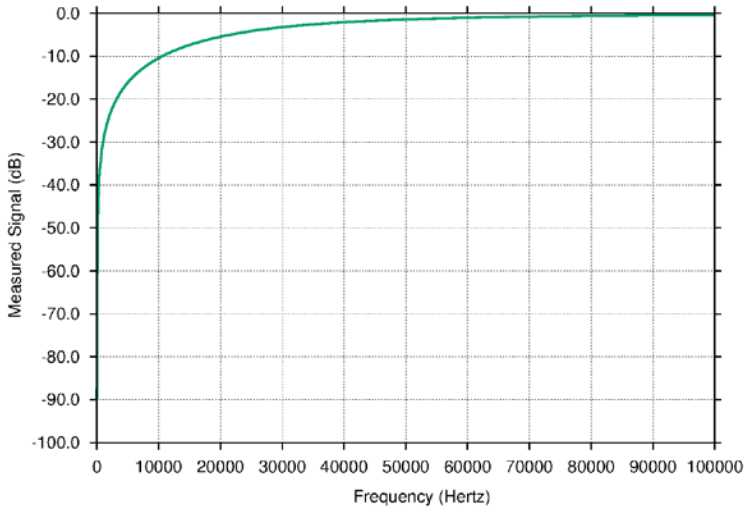

Figure 9 - A plot of the frequency response of the OpenADC input circuit as delivered.

Initially, DC coupling was desired for the input portion of the OpenADC circuit, but DC coupling was not possible with the type of LNA and VGA used in the circuit. Therefore, the lower cutoff frequency was minimized at around 5 or 6 Hz. Therefore, the capacitor on the input line was changed to 47 µF, as were the capacitors between the output of the VGA and the input to the ADC. Figures 11 & 12 show the frequency response plot of the model and the modeled circuit respectively. Subsequently, however, it has been determined that AC coupling with a lower cutoff frequency something like 4 or 5 times that of building AC (60 Hz) or 240 to 320 Hz, and even as high as 1 kHz, would be far better for the acquisition board since a lower cutoff at those numbers would help filter out facility electrical noise.

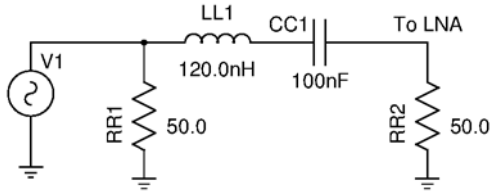

Figure 10 - A schematic of the input portion of the OpenADC circuit as delivered.

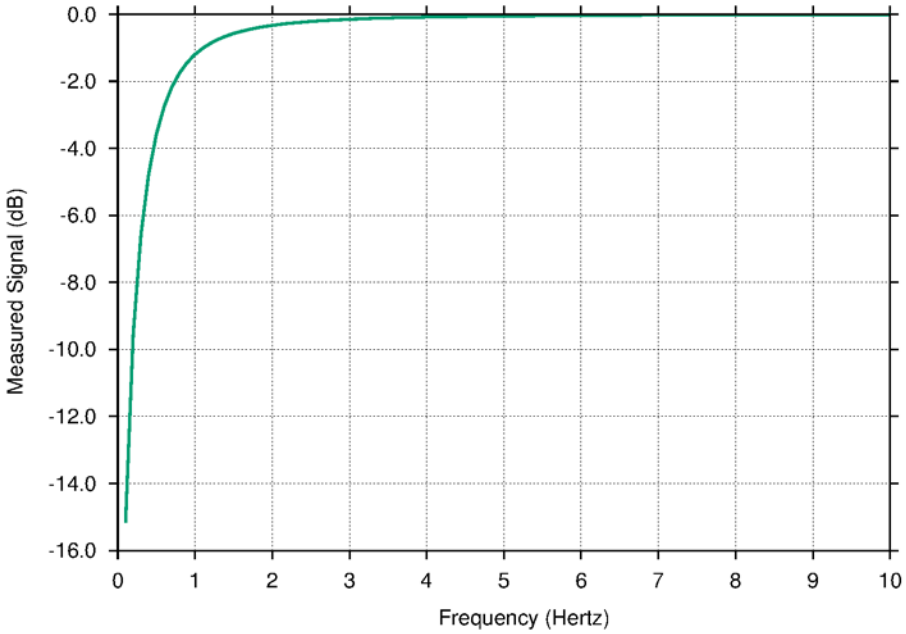

Figure 11 - A plot of the frequency response of the OpenADC input circuit as modified.

Finally, the jumper JP3 has been removed from the OpenADC board to increase the input impedance to about 6 KΩ. With the jumper in place, additional circuits became active that would "track" impedance to maintain the input impedance at 50 Ω. Ideally, the input impedance would be closer to 1 MΩ, but for now this compromise works. Figure 13 shows data digitized using one of the channels in the 4 channel chassis. The pulse was created with an Agilent 33220A Arbitrary Waveform generator set to square pulse generation into a high Z load.

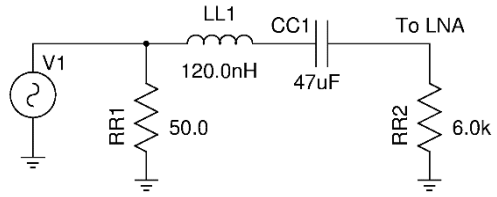

Figure 12 - A schematic of the input portion of the OpenADC circuit as modified.

Conclusion

A "trade study" was performed to find the most cost effective hardware and electronics for building a custom Beam Current Monitor Digital Acquisition System. Requirements were determined ahead of time such that products that could meet the requirements would stand out immediately. Cost was only one of the factors used for determining the hardware and electronics to use. The OpenADC board along with the Digilent Nexys 3 FPGA development board were chosen as hardware for a 4 channel BCM Digital data Acquisition system prototype.

The OpenADC 10-Bit, 100 MS/s analog to digital conversion board with a built-in, programmable, Low Noise Amplifier and Variable Gain amplifier, has been implemented in concert with a Digilent Nexys 3 FPGA development board to provide a custom, robust, and

inexpensive digital data acquisition device. Four of these assemblies, controlled with a Raspberry Pi 2, provides 4 channels of digitized data in an autonomous chassis. Workstations, servers, laptops, and even authorized tablets and phones can access the data or controls from anywhere on the same network as the acquisition chassis.

Changes were made to the factory OpenADC circuits such that they are better suited to the application of Beam Current Monitor Acquisition System. In particular, lower cutoff frequency has been lowered such that some of the "slower" signals, such as are expected in a system such as this, are not adversely affected by the circuitry. Since all of the circuit board schematics, Gerber files, etc. were made available with the OpenADC board, making changes to the design and manufacturing custom versions of the board are possible.

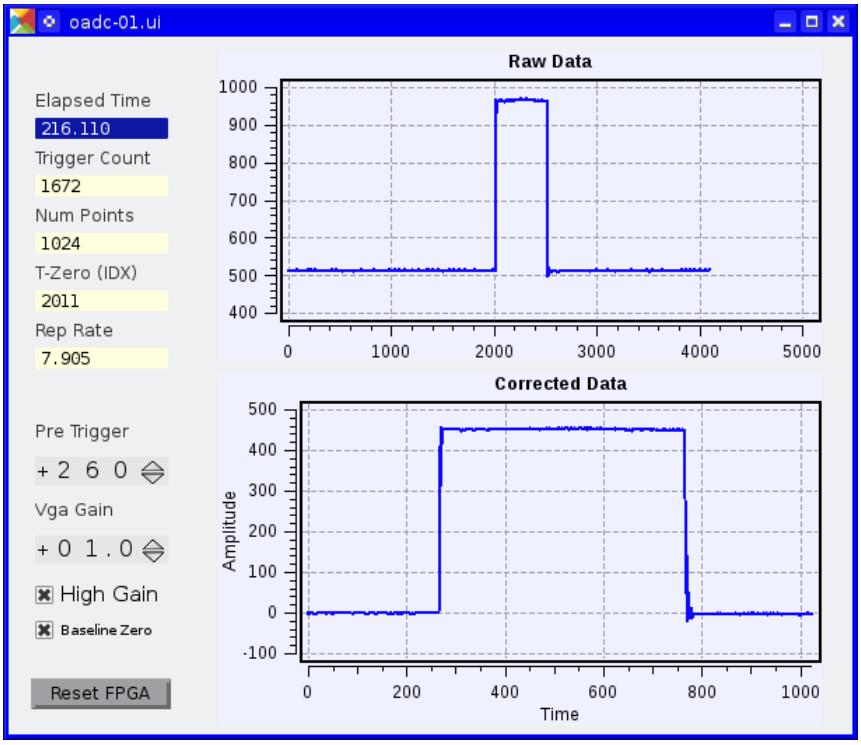

Figure 13 - Digitized and displayed data from one of the four channels.

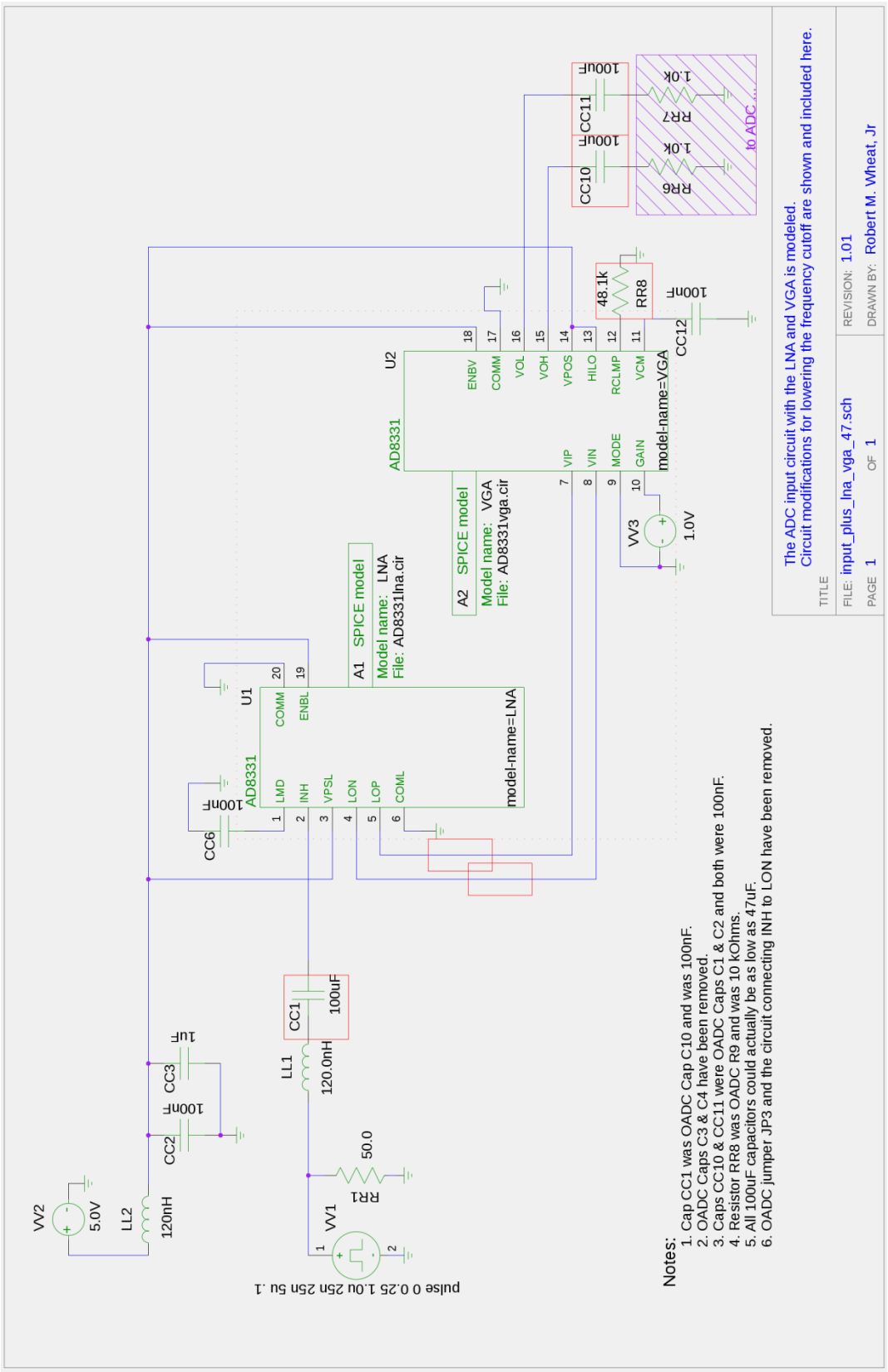

Figure 14 - A schematic showing changes made to the OpenADC board after purchase.

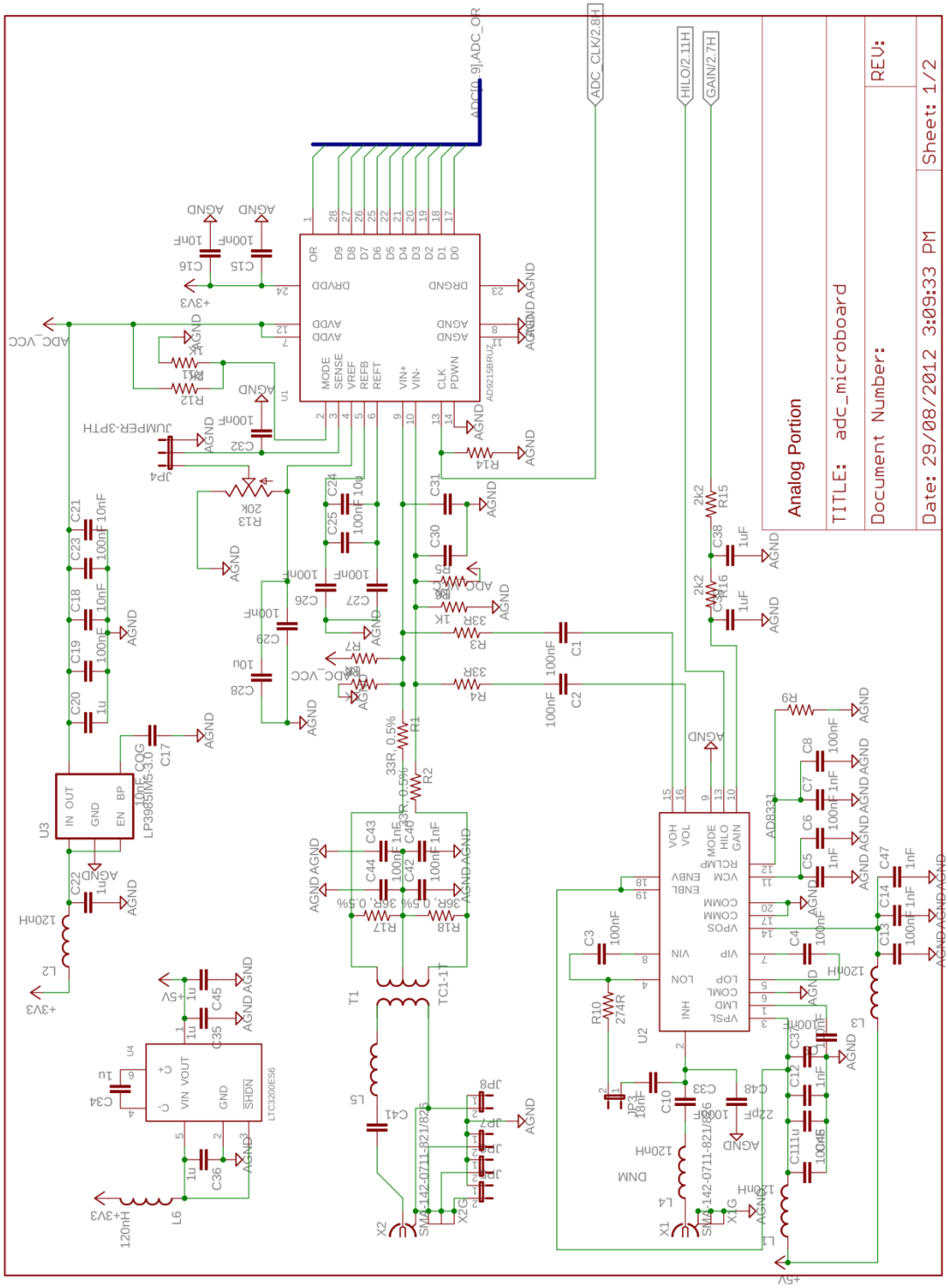

Figure 15 - OpenADC factory schematic.